

# OOSD Presentation

**Liam Scanlon 3727321**

**Shajie Chen 3582098**

**Sheng Kai Chen 3523185**

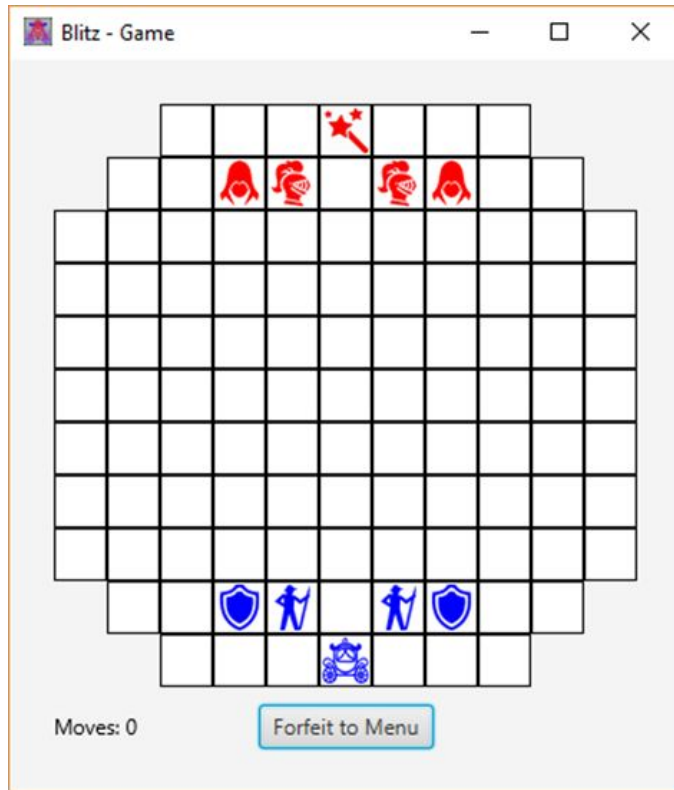
**Priyanga Dhilip Kumar 3670781**

# Blitz Game!

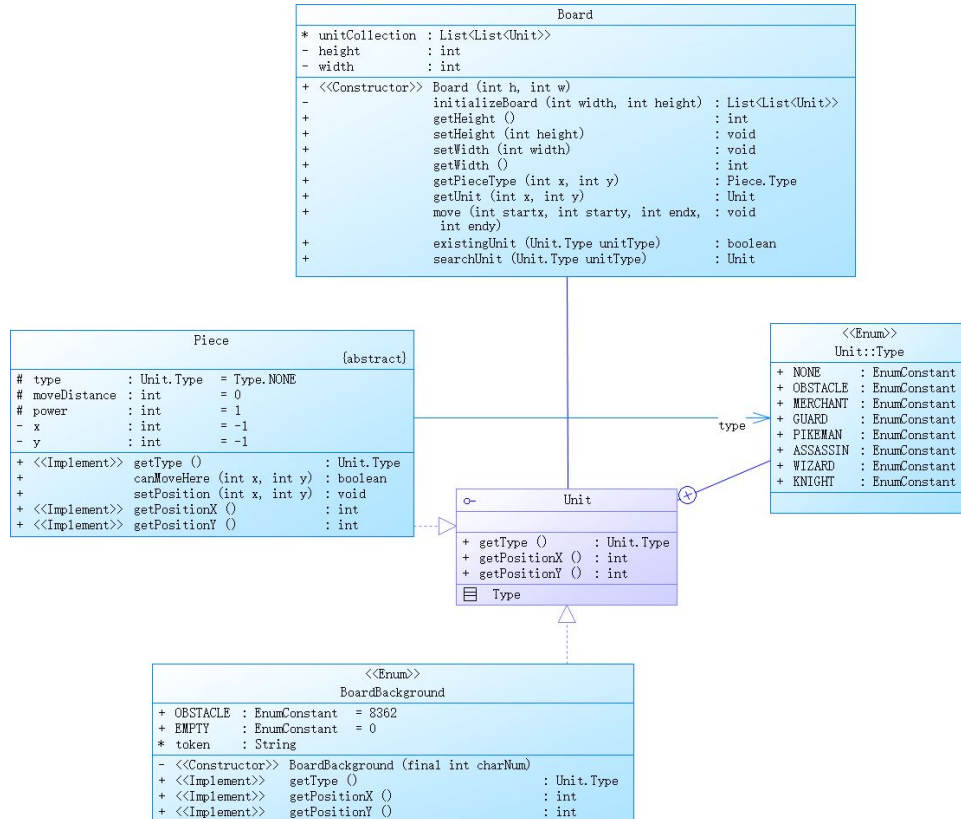
- Blue Team
  - Gold Merchant needs to get to the top of the board.
- Red Team
  - Needs to stop the Gold Merchant.

Pieces can only capture enemy pieces with an equal or lower power level. Each piece type has different movement distances, power levels, and some specific abilities:

	Move	Level	
Gold Merchant	2	0	(Carries the gold)
Guard	3	1	(Can take two hits)
Pikeman	2	3	(Can range attack 1 square away)
Assassin	4	1	(No obstacle penalty)
Wizard	2	2	(Can range attack 1 square away)
Evil Knight	3	3	(No ability)



# Class Diagram 1



## Polymorphism

The compile type is the Unit, but the runtime type is specific type. It prevents the duplicate code.

## Creator Pattern

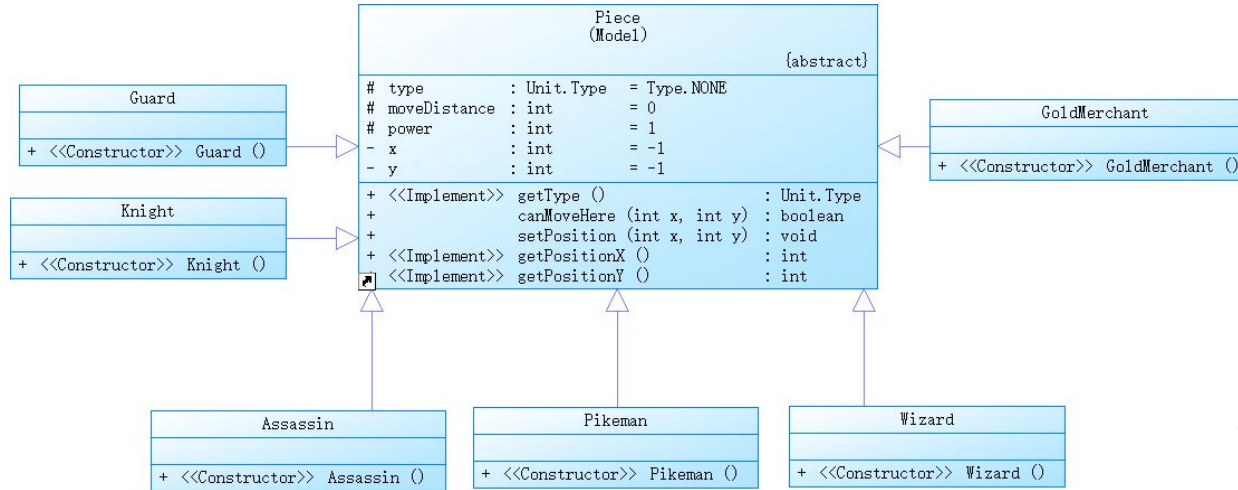
The board has a composition relationship with the Unit.

Unit interface is a square on the board, can be:  
BoardBackground  
Piece

## Single Responsibility

Board class has all its services narrowly aligned with the responsibility of managing the board.

# Class Diagram 2



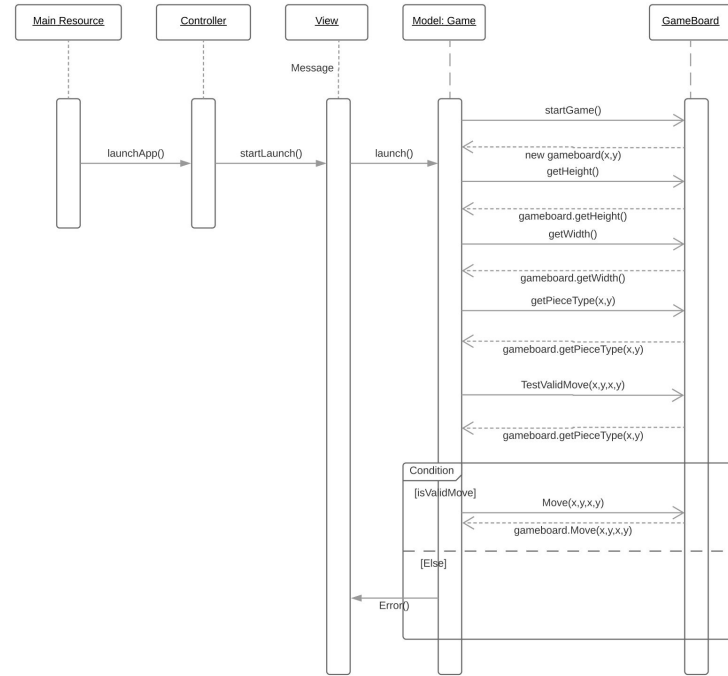
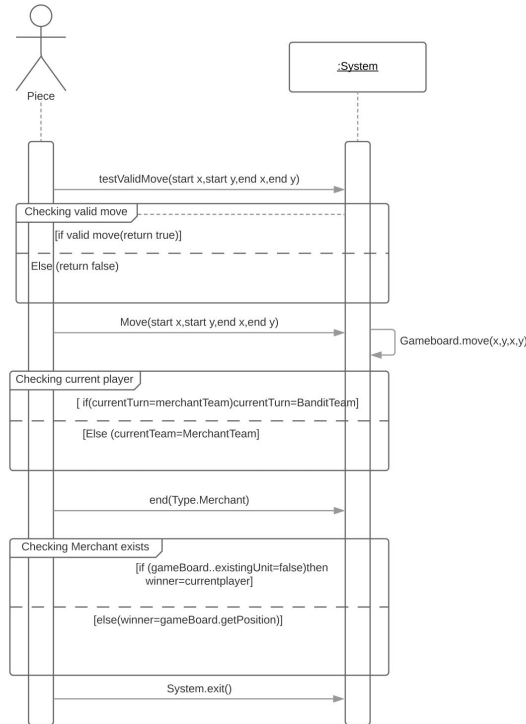
## Open-closed principle

The Pieces abstract class is not closed to modification when extending with another Piece. It is good for us when we need more different pieces.

## Liskov substitution

The specific piece will always work when treated as piece abstract class

# Sequence Diagrams



# Activity Diagram

